



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/741,860	12/22/2000	David Carlton Moore	088305/0129	8558

7590 07/13/2006

William T. Ellis
FOLEY & LARDNER
Washington Harbour
3000 K Street, N. W., Suite 500
Washington, DC 20007-5109

EXAMINER

ZHEN, LI B

ART UNIT PAPER NUMBER

2194

DATE MAILED: 07/13/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/741,860	MOORE ET AL.	
	Examiner	Art Unit	
	Li B. Zhen	2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 28 September 2005 and 31 October 2005.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-24 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 19 is/are allowed.
- 6) ☒ Claim(s) 1-12, 14-18, 20, 21, 23 and 24 is/are rejected.
- 7) ☒ Claim(s) 13 and 22 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

William Thomson
WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Claims 1 – 24 are pending in the current application.

Allowable Subject Matter

2. Claim 19 is allowed.
3. Claims 13 and 22 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.
4. The following is a statement of reasons for the indication of allowable subject matter:

The prior art of record does not expressly teach or render obvious the invention as recited in independent claim 19 and dependent claims 13 and 22 including all of the limitations of the base claim and any intervening claims.

The prior art discloses a interfacing between front-end data processing systems and back-end data processing systems [p. 31, figure in section 3, Client Layer and Database/Existing Applications of Mehra], an engine receiving from a front-end system a message incorporating a request for a transaction to be performed by a back end system [p. 32, section 3.2.1 of Mehra], the engine using a process map [configuration pattern; col. 7, lines 6 – 33 of Abjanic] to select one of a plurality of nodes in a node layer which may provide a suitable link to the back-end systems for the request [col. 7, lines 23 – 33 of Abjanic]. However, the prior art does not disclose the engine using a hashtable which maps keys to values for passing data and control to each other and passing the hashtable as a parameter in an execute method, a commit method, and a rollback method of a node object.

In addition, the prior art of record does not provide a basis of evidence for asserting a motivation that one of ordinary skill level in the art at the time the invention was made would have integrated or modified the interface for interfacing between front-end systems and back-end data processing systems to incorporate the features of using a hashtable which maps keys to values for passing data and control to each other and passing the hashtable as a parameter in an execute method, a commit method, and a

Art Unit: 2194

rollback method of a node object as recited in the context of independent claim 1 and the context of dependent claims 13 and 22 including all of the limitations of the base claim and any intervening claims.

Response to Arguments

5. Applicant's arguments filed 04/24/2006 have been fully considered but they are not persuasive.

In response to the Non-Final Office Action dated 01/23/2006, applicant argues that Abjanic does not teach that the engine uses the exposed node business logic to automatically build (and update) a process map (which links received messages to specific nodes) [p. 9, lines 5 – 18 and p. 10, lines 2 - 9].

Examiner respectfully disagrees. Abjanic discloses automatically generating [configuration patterns may be dynamically changed or updated by a program or application; col. 6, lines 50 – 63 of Abjanic] a process map [configuration pattern; col. 7, lines 6 – 33]. The process map linking received request messages with nodes, wherein the engine uses the process map to select the relevant node from the plurality of nodes because the director uses the configuration patterns to select a server to send the message to [col. 7, lines 23 – 33]. Clearly, Abjanic discloses a process map that is automatically created by a program or application and the process map is used to select a relevant node from a plurality of nodes [the servers as disclosed in Abjanic would correspond to the claimed nodes]. As to generating the process map based on exposed business logic capabilities of the nodes, Abjanic discloses that the configuration patterns identifies a plurality of servers and defines the business logic capabilities of each server. For example, server S1 has the capability to handle bookstore business transactions, server S2 has the capability to handle stock quote business transactions and server S3 has the capability to handle computer purchase orders [col. 7, lines 15 – 50]. The program or application that generates and updates the configuration patterns would need information from the servers [i.e. identity of the server and the capability of the server] in order to maintain the configuration patterns and ensure that transaction messages are properly directed to servers that are capable

Art Unit: 2194

of handling the transaction message. The servers exposes its business logic capabilities when they send messages to the program that maintains the configuration patterns. Therefore, the combination of Mehra and Abjanic teaches all the limitations as recited in the rejected claims.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. **Claims 1 – 12, 15 – 18, 20, 21, 23 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over “Designing a flexible services-based architecture for Internet Applications” [hereinafter Mehra] in view of U.S. Patent No. 6,732,175 to Abjanic, both references cited in the previous office action.**

8. As to claim 1, Mehra teaches the invention substantially as claimed including an interface for interfacing between front-end data processing systems and back-end data processing systems [p. 31, figure in section 3, Client Layer and Database/Existing Applications], the interface comprising an engine [p. 31, figure in section 3, Context Processor], a node layer comprising at least one node [p. 31, figure in section 3, Business Logic Layer], and a utility layer comprising at least one utility [p. 31, figure in section 3, Interface], and in which:

the engine configured to receive a message containing a request from a front-end system for a transaction to be performed by a back-end system [p. 32, section 3.2.1],

each node represents business logic interfaces to a back-end system [p. 32, section 3.3.1],

each node exposes business logic capabilities to the engine [p. 32, the interface of the Business Controller Object, section 3.3.2];

each utility is coupled as a proxy to a back-end system, and is configured to for receiving a transaction request from a node, for converting said request to a back-end system request, for receiving a response from the back-end system, and for routing a response to the requesting node [p. 32, sections 3.4.1 and 3.4.2],

each node routes a received response to the engine [p. 32, section 3.2.1 “The business logic layer...returns an XML output data stream”]; and

the engine routes a response to the requesting front-end system [p. 32, section 3.2.1 “The style sheet rendering engine merges the XML data stream with the appropriate XSL template file to render HTML output to the browser.”].

9. Although, Mehra teaches the invention substantially, Mehra does not specifically disclose the engines comprising means for interpreting the message to select a relevant node from a plurality of nodes for interfacing, wherein the engine does not contain any business logic and the message not containing an operation identifier of the transaction to be performed.

However, Abjanic teaches interfacing between front-end data processing systems [clients, for example, may include a server 110 that includes an application program 112, a computer 120 (such as a personal computer or laptop) that may include a web browser 122 and a wireless device 132, such as a personal digital assistant (PDA) or a wireless (or cellular) telephone; col. 3, lines 47 – 65] and back-end data processing systems [data center 135 includes several processing nodes (e.g., servers), including server 150, server 160 and server 170 for handling the various orders, business transactions and other requests; col. 3, line 65 – col. 4, line 12], an engine configured to receive a message containing a request from a front-end system for a transaction to be performed by a back-end system [content based message director (or XML director) can receive and switch messages based upon application data or business transaction information regardless of the transport or protocol used to transport the message; col. 3, lines 38 – 47; col. 4, lines 50 – 60], and means for interpreting said message to select a relevant node from a plurality of nodes for

interfacing [application data or XML data (including markup characters and/or character data) is then compared to one or more configuration patterns or queries (which may be stored in the director 145) to determine if there is a match; col. 6, lines 50 – 63] the message not containing an operation identifier of the transaction to be performed [a match is found between the business transaction information and the pattern, the content based switching logic 316 outputs a switching decision to a output interface 320. The switching decision may, for example, indicate that a match was found and identify the processing node or server; col. 9, lines 29 – 35 and col. 3, lines 14 – 23] wherein the engine does not contain any business logic [message director 145 to direct or switch messages to a selected server based upon the content of application data, such as business transaction information, col. 4, lines 50 – 60; examiner notes that the message director sends messages to the server or processing node to process the message, col. 3, lines 10 – 22, therefore, the message director does not contain any business logic], each node represents business logic interfaces to a back-end system, each node exposes business logic capabilities to the engine [different types of services (or different levels of service) to be provided for messages based on the content of the application data; col. 7, lines 33 – 43];

the engine comprises means for using the exposed node business logic capabilities to automatically build [configuration patterns may be dynamically changed or updated by a user or by a program or application. For example, a program may detect the failure of one or more servers and/or detect the response time of servers, and then update the configuration pattern to account for these changes in the network; col. 6, lines 50 – 63] a process map [configuration pattern; col. 7, lines 6 – 33] linking received request messages with nodes, wherein the engine uses the process map to select the relevant node from the plurality of nodes [Based on the above configuration patterns, the director 145 would direct a message to server; col. 7, lines 23 – 33]; and

each utility is coupled as a proxy to a back-end system, and is configured to for receiving a transaction request from a node, for converting said request to a back-end system request, for receiving a response from the back-end system, and for routing a

Art Unit: 2194

response to the requesting node [Broker 410 performs address translation on the header(s) for the message; col. 10, lines 33 – 67].

10. It would have been obvious to a person of ordinary skill in the art at the time of the invention to apply the teaching of interpreting the message to select a relevant node from a plurality of nodes for interfacing, wherein the engine does not contain any business logic and the message not containing an operation identifier of the transaction to be performed as taught by Abjanic to the invention of Mehra because this offloads the burden of examining the application data or business transaction information and then switching to a particular processing node from application servers [col. 3, lines 33 - 39 of Abjanic].

11. As to claim 2 Mehra as modified teaches the engine comprises means for dynamically maintains the process map according to the exposed node business logic capabilities [col. 6, lines 50 – 63 of Abjanic].

12. As to claim 3, Mehra teaches the process map comprises a script file [XML is used for structured data message; Section 2, p. 31].

13. As to claim 4, Mehra as modified teaches the process map comprises script messages, each message having a map associating incoming parameter names with standardised names [col. 7, lines 23 – 33 of Abjanic].

14. As to claim 5, Mehra as modified teaches each message of the process map specifies an associated node, a list of the parameters the node requires, and values which it returns for a type of incoming message [col. 7, lines 13 – 33 of Abjanic].

15. As to claim 6, Mehra teaches that the utilities interface with the node layer according to a uniform interface model [“...that provide an abstraction layer that encapsulates access...”; section 3.3.2, p. 32].

Art Unit: 2194

16. As to claim 7, Mehra as modified teaches the engine calls a plurality of nodes for a transaction request [col. 7, lines 2 – 6 of Abjanic].

17. As to claim 8, Mehra as modified teaches the engine is configured for calling nodes in sequence, and for passing the output from a previous node to a next node [col. 9, lines 50 – 55 of Abjanic].

18. As to claim 9, Mehra as modified teaches the engine and each node uses a hashtable mapping keys to values for passing data and control to each other [col. 10, lines 33 – 67 of Abjanic].

19. As to claims 10 and 11, Mehra as modified teaches the engine and the nodes each use a hashtable for returning a result from a back-end system [col. 10, lines 33 – 67 of Abjanic].

20. As to claim 12, Mehra teaches each of the engine and each node comprise an object instantiated from an object-oriented class [section 3.3.2, p. 32].

21. As to claim 15, Mehra as modified teaches the engine comprises an externally visible engine class ["Servlet"; section 3.2.2 "Servlet" of Mehra], an object of which comprises means for instantiating: a processor object for instantiating said node objects ["Business Object Factory"; section 3.3.1, p. 31 of Mehra]; and a loader object for loading the process map, and for determining node objects associated with a received message [col. 6, lines 50 – 63 of Abjanic].

22. As to claim 16, Mehra as modified teaches the engine is configured for instantiating a parser object for parsing a received message [parsing all or part of the application data; col. 6, lines 39 – 49 of Abjanic], for placing extracted data in a hashtable, and for returning the hashtable to the engine object [col. 10, lines 33 – 67 of Abjanic].

23. As to claim 17, Mehra as modified teaches the engine comprises a builder object configured for automatically updating the process map according to capabilities exposed by node classes [col. 6, lines 50 – 63 of Abjanic].

24. As to claim 18, Mehra as modified teaches each node class comprises a method for returning a string to the engine indicating the node capabilities [program may detect the failure of one or more servers and/or detect the response time of servers, and then update the configuration pattern to account for these changes in the network; col. 6, lines 56 – 63 of Abjanic].

25. As to claim 20, Mehra as modified teaches a method of interfacing between front-end data processing systems and back-end data processing systems [p. 31, figure in section 3, Client Layer and Database/Existing Applications of Mehra], the method being performed by an interface comprising an engine [p. 31, figure in section 3, Context Processor of Mehra] for communicating with the front-end systems and a utility layer for communicating with the back-end systems, the method comprising the steps of:

the engine receiving from a front-end system a message incorporating a request for a transaction to be performed by a back end system but not indicating a particular back-end system suitable for the transaction [p. 32, section 3.2.1 of Mehra], wherein the message does not include an operation identifier identifying the transaction to be performed [a match is found between the business transaction information and the pattern, the content based switching logic 316 outputs a switching decision to a output interface 320. The switching decision may, for example, indicate that a match was found and identify the processing node or server; col. 9, lines 29 – 35 and col. 3, lines 14 – 23 of Abjanic; examiner notes the identity of the processing node is determined by the switching logic and not the message],

the engine using a process map [configuration pattern; col. 7, lines 6 – 33 of Abjanic] to select one of a plurality of nodes in a node layer which may provide a suitable link to the back-end systems for the request [col. 7, lines 23 – 33 of Abjanic],

Art Unit: 2194

the process map being automatically built [configuration patterns may be dynamically changed or updated by a user or by a program or application. For example, a program may detect the failure of one or more servers and/or detect the response time of servers, and then update the configuration pattern to account for these changes in the network; col. 6, lines 50 – 63 of Abjanic] to link message types to nodes according to exposed business logic capabilities of the nodes [col. 7, lines 33 – 43 of Abjanic], wherein the engine does not contain any business logic [message director 145 to direct or switch messages to a selected server based upon the content of application data, such as business transaction information, col. 4, lines 50 – 60 of Abjanic; examiner notes that the message director sends messages to the server or processing node to process the message, col. 3, lines 10 – 22 of Abjanic, therefore, the message director does not contain any business logic],

the engine passing a request to the selected node [p. 32, the interface of the Business Controller Object, section 3.3.2 of Mehra],

the selected node communicating with a utility with which it is associated to instruct the utility to perform the transaction, receiving a response from the utility, and passing the response back to the node [p. 32, sections 3.4.1 and 3.4.2 of Mehra],

the node passing the response back to the engine [p. 32, section 3.2.1 “The business logic layer...returns an XML output data stream” of Mehra], and the engine passing the response back to the requesting front-end [p. 32, section 3.2.1 “The style sheet rendering engine merges the XML data stream with the appropriate XSL template file to render HTML output to the browser” of Mehra].

26. As to claim 21, this is rejected for the same reasons as claims 10 and 15 above.

27. As to claim 23, Mehra teaches the process map is an XML script file [section 3.1.1, p. 31].

28. As to claim 24, this is a product claim that corresponds to method claim 20; note the rejection to claim 20 above, which also meets this product claim.

29. **Claim 14 is rejected under 35 U.S.C. 103(a) as being unpatentable over Mehra and Abjanic further in view of U.S. Patent No. 6,434,555 to Frolund et al. [hereinafter Frolund, cited in previous office action].**

30. As to claim 14, Mehra as modified does not teach the engine is configured for activating a sequence of nodes for a transaction, and each node is configured for performing a rollback if a transaction fails.

However, Frolund teaches the engine is configured for activating a sequence of nodes for a transaction, and each node is configured for performing a rollback if a transaction fails [col. 4, lines 21 - 52 of Frolund].

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine Mehra, Abjanic and Frolund because this provides outcome determination so that the result of a transaction can be used to make a decision [col. 2, lines 28 – 39 of Frolund].

Conclusion

31. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

CONTACT INFORMATION

Art Unit: 2194


32. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (571) 272-3768. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Thomson can be reached on 571-272-3718. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Li B. Zhen
Examiner
Art Unit 2194

lbz


WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER